

Prompt não é mágica: o Ciclo de Remediação Ativa como estratégia de formação algorítmica crítica na era da IA

Marcelo Magalhães Foohs¹, Simone Rosanelli Dullius², Patrícia Campelo Costa Barcellos³

¹Departamento de Estudos Especializados
Universidade Federal do Rio Grande do Sul (UFRGS) – Porto Alegre, RS – Brazil

²Centro Interdisciplinar de Novas Tecnologias na Educação
Universidade Federal do Rio Grande do Sul (UFRGS) – Porto Alegre, RS – Brazil

³Instituto de Letras
Universidade Federal do Rio Grande do Sul (UFRGS) – Porto Alegre, RS – Brazil

mmfoohs@gmail.com, simone-rdullius@educar.poa.br,
patricia.campelo@ufrgs.br

Abstract. *This essay analyzes the Active Remediation Cycle (ARC) as a formative proposal that integrates multiliteracies, programming, and computational thinking, drawing on authors such as Papert, Resnick, Rojo, Koch, and Faraco. The ARC fosters learners' ability to design algorithms with critical authorship while simultaneously enhancing their linguistic competence across different varieties of the Portuguese language. The argument is supported by reflections drawn from responses to a questionnaire administered to sixteen elementary school teachers after a three-month training program, revealing perceptions about learning processes, challenges, and the appropriation of the ARC as a critical language of authorship in times of generative artificial intelligence.*

Resumo. *Este ensaio analisa o Ciclo de Remediação Ativa (CRA) como proposta formativa que integra multiletramentos, programação e pensamento computacional, com base em autores como Papert, Resnick, Rojo, Koch e Faraco. O CRA desenvolve nos sujeitos a capacidade de planejar algoritmos com autoria crítica e, ao mesmo tempo, aprimorar sua competência linguística em diversas variedades da língua portuguesa. A argumentação encontra suporte nas respostas a um questionário aplicado a dezesseis professores do ensino fundamental após formação de três meses, revelando percepções sobre aprendizagem, dificuldades e apropriação do CRA como linguagem crítica de autoria em tempos de inteligência artificial generativa.*

1. Introdução

Nos últimos anos, a formação docente em Computação tem enfrentado uma tensão epistemológica cada vez mais evidente: de um lado, o esforço para consolidar uma abordagem crítica, autoral e situada da programação; de outro, a sedutora promessa de que bastará escrever comandos em linguagem natural para que a máquina faça tudo. Essa promessa, intensificada pela ascensão das inteligências artificiais generativas, alimenta discursos tecnodeterministas que anunciam o fim da necessidade de aprender a

programar. Afinal, se um *prompt* resolve tudo, por que investir em estruturas algorítmicas?

Essa visão, embora atraente em sua superfície, carrega o risco de desmobilizar justamente aquilo que mais importa na educação computacional: o desenvolvimento da consciência algorítmica, da autoria crítica e da capacidade de pensar o próprio pensamento (Papert, 1980). Escrever um *prompt*, ao menos no contexto de gerar jogos, narrativas interativas ou aplicações computacionais, não se trata de um mero improviso verbal, mas de planejamento expressivo, estratégico e consciente. Nesse cenário, o desenvolvimento do pensamento computacional não deve ser compreendido como um treinamento em frases mágicas, mas fortalecido como espaço de formação epistêmica e crítica. É nesse horizonte pedagógico que se insere o Ciclo de Remediação Ativa (CRA), uma proposta que integra multiletramentos e pensamento computacional a partir de um percurso que inclui leitura crítica, reescrita autoral, planejamento algorítmico, prototipagem, implementação e metacognição.

Assim, neste ensaio analisaremos o CRA como uma estratégia de educação algorítmica crítica para tempos de IA generativa no contexto de formação continuada. Sem desconsiderar a relevância dos estudos sobre engenharia de *prompts*, propõe-se situar os *prompts* como resultados de um planejamento algorítmico consciente, e não como fórmulas improvisadas. O foco, portanto, não está no ensino técnico de elaboração de *prompts*, mas na compreensão de que interações qualificadas com sistemas generativos decorrem de processos prévios de leitura crítica, reescrita e planejamento lógico.

A argumentação encontra suporte em respostas provenientes de um questionário aplicado a 16 professores articuladores do ensino fundamental que participaram de uma formação continuada de três meses, promovida no primeiro semestre de 2025. Essa formação, cuidadosamente estruturada em torno das etapas do CRA, teve como eixo integrador a remediação de fábulas em ambientes de programação visual, como o Scratch, em diálogo com os fundamentos da BNCC (2018), da Resolução CNE/CEB n. 1/2022 e das diretrizes da BNCC Complementar de Computação (2022). Além das oficinas práticas, o plano contemplou momentos de fundamentação teórica, análise de exemplos e discussões coletivas que possibilitaram aos docentes vivenciar o CRA como suporte para autoria e mediação crítica. Durante o processo, ferramentas de IA generativa foram utilizadas de maneira pontual, sobretudo como apoio na resolução de problemas específicos de implementação e na geração de imagens e personagens. Seu uso ocorreu sempre articulado ao planejamento algorítmico, no interior de um percurso formativo de natureza iterativa. O recurso à IA, quando acionado, não suprimiu a necessidade de domínio lógico; ao contrário, evidenciou-a. Formular a pergunta adequada e compreender a resposta gerada exigiam intencionalidade algorítmica, além do conhecimento da lógica do ambiente de programação utilizado. A IA, portanto, não dispensou o processo formativo, mas o pressupôs.

É significativo que, nas respostas finais, a IA simplesmente não tenha sido mencionada. Tal ausência sugere que sua presença foi naturalizada como ferramenta de apoio, e não percebida como eixo estruturante da experiência formativa. Assim como não se comenta o uso de um editor de texto ou de imagem, a IA foi incorporada como artefato de apoio à autoria, e não como delegação de autoria, permanecendo o foco formativo no processo de reescrita, algoritmização e reflexão crítica. Suas percepções

sobre aprendizagem, dificuldades e apropriação do ciclo, registradas ao final do percurso, revelam tanto desafios técnicos quanto posicionamentos epistêmicos diante da inserção crítica da Computação nas aprendizagens do ensino básico.

Essa análise articula-se, ainda, com propostas internacionais como a de *Computational Empowerment* (Dindler et al., 2022), que concebem a educação computacional não como mera aquisição de habilidades técnicas, mas como processo formativo de agência crítica e design participativo. Assim, o CRA insere-se nesse ecossistema de abordagens que valorizam a autoria, a criticidade e a mediação situada na aprendizagem computacional.

2. O Ciclo de Remediação Ativa: fundamentos e percurso

Concebido inicialmente no contexto da pesquisa de Foohs e Giraffa (2022), o Ciclo de Remediação Ativa (CRA) nasce do desejo de reconciliar campos historicamente apartados na prática escolar: linguagem e computação. Diferentemente de abordagens que tratam a programação como um apêndice técnico ao currículo, o CRA propõe que os saberes computacionais sejam incorporados de forma integral, com gramática, retórica e estilo próprios, isto é, como linguagem plena de autoria e expressão, não apenas como ferramenta técnica, mas como forma de expressão significativa, modo de pensar e ação intencional.

Nesse sentido, torna-se urgente distinguir propostas que preservam essa complexidade epistêmica de outras que reduzem o pensamento computacional a atividades genéricas ou desvinculadas do rigor algorítmico. Como argumentam Foohs, Rebouças e Krüger (2025), a adoção do pensamento computacional como competência transversal na BNCC não deve ser confundida com exercícios criativos descontextualizados, ou com o mero seguimento de sequências sem intencionalidade lógica. Tal alerta reforça o papel de abordagens como o CRA, que integram linguagem, lógica e autoria em um mesmo gesto formativo.

A base teórica que sustenta o CRA é plural. Papert (1980), ao propor a ideia de que as crianças podem tornar-se epistemólogas de seu próprio processo de aprendizagem, oferece o solo fértil sobre o qual repousa a noção de que aprender a programar não é apenas dominar comandos, mas construir formas de pensar o mundo com mais clareza, estrutura e potência transformadora. Resnick (2020), por sua vez, ao defender uma aprendizagem criativa, lúdica e relevante, contribui com a visão de que o CRA deve ser mais do que uma sequência de etapas: ele precisa cultivar ambientes em que os alunos experimentem, brinquem, projetem e compartilhem, tal como se faz em um jardim de infância para a vida toda.

A influência de Rojo (2012) é igualmente decisiva. Ao tratar os multiletramentos como ferramentas para inclusão social e valorização das diversas formas de expressão e representação, Rojo (2012) legitima a ideia de que trabalhar com algoritmos também é trabalhar com letramentos, letramentos lógicos, visuais, narrativos e computacionais, que precisam ser mediados de forma crítica. Assim, o CRA se torna um espaço em que múltiplas linguagens se entrelaçam, incluindo a linguagem da programação como uma linguagem da autoria e da imaginação.

Nesse sentido, a proposta de “remediação”, entendida conforme Bolter e Grusin (2000) como o processo de reinscrever e transformar uma mídia em outra, fornece o enquadramento conceitual para compreender o movimento essencial do CRA: transpor

uma narrativa escolar (oral, escrita ou imagética) para uma forma digital e interativa, preservando sua intencionalidade sem renunciar ao potencial expressivo das mídias computacionais. Tal gesto não é meramente técnico, é epistemológico. É o que exige que professores e alunos se reconfigurem como produtores, e não apenas como consumidores de tecnologia.

Além disso, no CRA manifesta-se com nitidez a articulação entre linguagem, autoria e consciência crítica. A elaboração textual constitui dimensão obrigatória do processo formativo, uma vez que o CRA assume explicitamente o enfrentamento da atual carência de expressão na norma padrão como responsabilidade pedagógica. Essa obrigatoriedade, contudo, não implica organização linear do pensamento criativo: a emergência da ideia pode apresentar-se de modo imagético, difuso ou mesmo já inscrita em um protótipo computacional. A linguagem verbal não antecede necessariamente o gesto algorítmico, mas integra um campo de mediações em que narrativas se configuram entre diferentes materialidades semióticas. Nesse contexto, o trabalho com a norma culta realiza-se com profundidade, não como imposição hierárquica, mas como ampliação do repertório expressivo no interior da variação linguística. O CRA ancora-se, assim, em fundamentos da linguística crítica contemporânea ao compreender que formar programadores-autores é também formar sujeitos capazes de agir discursivamente com consciência de contexto, de destinatário e de repertório.

Como observa Faraco (2015, p. 20), com contundência:

Sobre variação linguística e ensino, não podemos mais nos contentar com generalidades. Dispomos já, como produto de décadas de reflexões e debates, de uma formulação geral com diretrizes que incorporam o estudo da variação linguística entre os temas do ensino de português e situam o trabalho com a expressão culta no interior do quadro mais amplo da variação linguística que caracteriza a nossa sociedade. Percebe-se, porém, que esta formulação geral, embora bem fundamentada nos nossos saberes mais técnicos sobre como a língua é e como ela funciona, não conseguiu ainda alterar substancialmente conceitos, atitudes e práticas, seja no geral da sociedade, seja no específico da educação.

Essa constatação, segundo Faraco (2015), revela o quanto práticas escolares ainda operam sob a lógica da “régua estreita do certo e do errado”, desconsiderando que é justamente na variação social, e não apenas geográfica, que se produzem os gestos de exclusão, preconceito e violência simbólica. O CRA, ao estimular a produção de textos em múltiplas linguagens e suportes, convoca os alunos a refletirem sobre o que dizem e como dizem, para quem dizem e com que intencionalidade discursiva. Essa abordagem se conecta diretamente à noção de situacionalidade textual (Koch, 2020), que desloca a ênfase da norma para a adequação comunicativa contextual, pilar fundamental para qualquer narrativa, seja ela impressa, interativa ou algorítmica.

Nessa esteira, a necessidade do aprimoramento da língua portuguesa torna-se ainda mais evidente quando considerada à luz dos resultados do SAEB (Brasil, 2024), que evidenciam o baixo desempenho dos estudantes brasileiros em competências de leitura e escrita, especialmente no uso da norma-padrão da língua portuguesa. O CRA, ao promover o trânsito entre diferentes registros linguísticos e ao valorizar o contexto comunicativo de cada produção, busca formar autores capazes de fazer escolhas conscientes de linguagem. Em vez de impor modelos, o ciclo propicia um espaço formativo em que linguagem e autoria se entrelaçam, respeitando os repertórios dos alunos e exigindo deles uma atuação como sujeitos comunicativos em diferentes esferas.

Com isso, o CRA articula planejamento computacional e letramento crítico como dimensões indissociáveis, ao tratar a linguagem como gesto situado e autoral, em que a seleção da variedade linguística se configura como decisão estratégica, estética e ética.

O CRA estrutura-se em seis dimensões articuladas: Produção Textual, Decomposição Narrativa, Construção de Algoritmo Narrativo, Implementação Digital, Depuração e Refinamento, e Socialização e Reflexão, que operam em dinâmica recursiva e iterativa, e não como etapas lineares. Trata-se de uma arquitetura metodológica em que diferentes dimensões do pensamento computacional e do letramento são continuamente mobilizadas, permitindo ao sujeito transitar entre criação, análise, experimentação e reflexão sobre sua própria produção.

No âmbito do CRA, a Produção Textual configura-se como espaço de exercício do gesto autoral e da organização discursiva; a Decomposição Narrativa favorece a análise estrutural do enredo e de seus elementos constitutivos; a Construção de Algoritmo Narrativo exige a formalização lógica das ações e relações; a Implementação Digital materializa a proposta em ambiente computacional; a Depuração e Refinamento promovem ajustes estruturais e discursivos; e a Socialização e Reflexão consolidam o retorno crítico ao processo, ampliando a consciência linguística, estética e ética do autor.

Essas dimensões não obedecem a uma ordem fixa de ativação, podendo ser retomadas, tensionadas e reorganizadas conforme as necessidades da criação, o que reafirma o CRA como metodologia de caráter cíclico e não como sequência didática linear.

No que se refere à prototipagem, é fundamental diferenciá-la de uma simples atividade de planejamento ou de roteirização abstrata. No CRA, a prototipagem constitui espaço de ensaio concreto do pensamento algorítmico, configurando-se como prática em que a ideia ganha materialidade provisória. Ela se manifesta, em muitos casos, por meio de encenações e expressões multimodais, nas quais o sujeito dramatiza, verbaliza ou representa graficamente o funcionamento de sua proposta. Essa dimensão expressiva constitui o próprio núcleo da prototipagem, pois permite que o pensamento adquira forma perceptível e possa ser relido pelo próprio autor. Quando essa materialização ocorre no meio digital, a implementação e a depuração de erros, a chamada caça aos *bugs*, convertem-se em leitura crítica do próprio raciocínio. O erro, nesse contexto, deixa de ser falha técnica e torna-se espelho metacognitivo, ao exigir que o sujeito retome e examine a lógica que produziu.

Ao longo das diferentes dimensões do CRA, a mediação docente e a mediação por artefatos culturais assumem papel central. Como destaca Vygotsky (1998, 2001), não há desenvolvimento pleno das funções psicológicas superiores sem a ação do outro e sem o uso de instrumentos simbólicos e materiais que ampliam o pensamento. O estudante, nesse processo, é convidado a assumir o papel de planejador textual-algorítmico, isto é, sujeito que articula a intencionalidade comunicativa da linguagem com a estrutura operacional da lógica computacional. A noção de planejamento, aqui, não constitui elemento acessório, mas o núcleo da aprendizagem. Como observa Koch (2020), o texto exige coesão, coerência e clareza estrutural, requisitos igualmente indispensáveis à construção de algoritmos, ainda que pertencentes a outra ordem de linguagem. A analogia entre produção textual e programação, portanto, não é meramente ilustrativa: ambos os processos demandam arquitetura, escolhas conscientes

e domínio dos efeitos de sentido. Ignorar essa dimensão estrutural em favor de soluções automatizadas e não refletidas é desconsiderar a própria natureza dos processos de criação.

A Figura 1 representa o CRA como uma arquitetura metodológica de natureza cíclica, em que as diferentes dimensões se articulam em fluxo iterativo e retroalimentado. Essa representação evita a rigidez de uma linearidade causal simples e enfatiza a natureza relacional e processual do percurso formativo. Cada dimensão, ao se conectar com as demais, configura um ciclo dinâmico, no qual leitura, escrita, planejamento, expressão e programação se entrelaçam continuamente, conduzindo a processos de metacognição que sintetizam provisoriamente a experiência e a reabrem para novas interpretações.

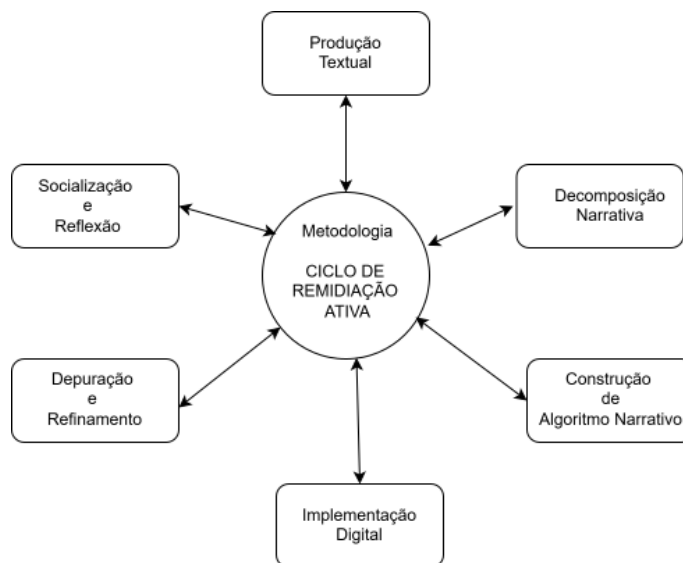


Figura 1. Ciclo de Remediação Ativa

Na Figura 1, a metacognição não se apresenta como ponto de chegada, mas como dimensão reflexiva que atravessa o Ciclo de Remediação Ativa. Trata-se do movimento pelo qual o estudante toma consciência do próprio processo de criação (da leitura à escrita, do planejamento à prototipagem e à expressão multimodal) compreendendo como pensamento linguístico e pensamento algorítmico se articularam em sua produção. Em vez de apenas concluir uma tarefa, o aprendiz observa suas escolhas, dificuldades e estratégias, identifica o que funcionou e o que poderia ser reorganizado e reconhece os deslocamentos de sentido que produziu. Essa autorreflexão constitui parte estruturante do CRA: ao pensar o próprio pensamento, o sujeito transforma o produto e a si mesmo, gerando novas leituras, novas escritas e novas formas de agir com linguagem e lógica. O ciclo, assim, não se encerra, mas se reconfigura continuamente em novos níveis de consciência.

Para que essa dimensão metacognitiva se consolide como componente constitutivo do CRA, a avaliação precisa ser compreendida como mediação interpretativa do processo, e não como veredito sobre o produto. Avaliar, nesse contexto, significa ler o percurso do aprendiz: observar como ele nomeia suas estratégias, reformula decisões e reconhece os deslocamentos que experimenta. Instrumentos como diários de bordo, narrativas reflexivas, rubricas dialógicas e rodas de conversa tornam visível o pensamento em movimento. Mais do que medir desempenho,

tais dispositivos permitem interpretar a consciência emergente, evidenciando o grau em que o sujeito passou a ler o próprio raciocínio com lucidez, intencionalidade e responsabilidade autoral.

3. A Era do *Prompt* e a Ilusão da Simplicidade Computacional

Vivemos uma virada paradigmática. O avanço das IAs generativas, especialmente os Large Language Models (LLMs), como o ChatGPT, lançou a promessa de que tarefas antes associadas ao domínio técnico da programação poderiam agora ser executadas por meio de uma simples frase: um *prompt*. A retórica publicitária que se dissemina em redes sociais, painéis corporativos e até entre educadores entusiasmados com a inovação, sugere que bastaria “escrever bem” para que a máquina fizesse o resto.

Essa crença, no entanto, carrega uma ilusão perigosa: a de que a complexidade do raciocínio computacional pode ser substituída por improvisação em linguagem natural. Ao contrário, aquilo que se apresenta como facilidade esconde, muitas vezes, uma demanda ainda mais sofisticada por clareza algorítmica. O *prompt* não é mágica. O *prompt* é, no fundo, um algoritmo expresso em linguagem natural e, como tal, carrega intencionalidade, estrutura, dependências e lógica de execução. A confusão entre linguagem natural e estrutura computacional não é nova, mas tem sido amplificada na era dos modelos generativos de linguagem. Com a ascensão de interfaces que traduzem comandos verbais em código, como as oferecidas por plataformas baseadas em LLMs, consolida-se uma retórica perigosa: a de que saber programar equivale a saber “pedir bem”. Trata-se de um deslocamento que simplifica indevidamente a noção de pensamento computacional, reduzindo-a a uma competência linguística informal, descolada dos fundamentos técnicos e epistêmicos da computação.

Contra essa tendência, diferentes documentos normativos e estudos especializados têm reiterado que o pensamento computacional não se resume à formulação de instruções em linguagem natural, mas envolve processos complexos de abstração, decomposição, reconhecimento de padrões e construção de algoritmos. Os padrões da CSTA (Computer Science Teachers Association, 2017), por exemplo, estruturam a aprendizagem da computação ao longo da educação básica com base em práticas progressivas de construção algorítmica e resolução de problemas, integrando estruturas como condicionais, laços, variáveis e modularização. A ênfase recai na capacidade de estruturar soluções computáveis de modo lógico, preciso e interpretável pela máquina, independentemente da linguagem de programação empregada.

No mesmo sentido, Denning e Tedre (2019; 2021) oferecem uma crítica à popularização acrítica do termo *Computational Thinking*, demonstrando que, ao longo da última década, o conceito foi frequentemente desfigurado por interpretações superficiais e instrumentalizadas. Para os autores, pensar computacionalmente é mais do que aplicar boas práticas de resolução de problemas: é operar dentro de uma disciplina com princípios, métodos e linguagens próprios, cuja essência está no domínio da representação, automação e transformação de processos informacionais. Eles argumentam que o pensamento computacional disciplinar não pode ser reduzido a *slogans* ou metáforas, sob pena de esvaziamento conceitual.

Ao se recuperar esse horizonte disciplinar, em que o desenvolvimento do estudante constitui o eixo formativo, evidencia-se que formulações improvisadas, ainda que operacionalmente eficazes, não promovem a internalização de estruturas

algorítmicas necessárias ao amadurecimento cognitivo. A produção de um *prompt* eficaz, nessa perspectiva, requer domínio do que poderíamos chamar de retórica algorítmica. Assim como ninguém se torna orador apenas por tagarelar com facilidade, ninguém se torna programador apenas por formular frases coerentes em linguagem natural. A fluência verbal, inclusive quando aplicada ao uso de comandos para uma IA, não garante profundidade estrutural, intencionalidade expressiva ou rigor lógico. Um bom orador domina não somente o vocabulário, mas a arquitetura da fala: introdução, desenvolvimento, clímax, conclusão, recursos retóricos, estratégias de envolvimento. Do mesmo modo, um bom *prompt*, quando destinado, por exemplo, à criação de uma calculadora, de um jogo ou de uma narrativa interativa, precisa mobilizar estruturas algorítmicas como repetições, condicionais, paralelismo, listas e variáveis, mesmo que essas estruturas estejam ocultas sob a superfície aparentemente natural da linguagem. A eloquência discursiva e a eficácia computacional partilham, portanto, uma mesma exigência: planejamento consciente. Improvisar um discurso pode até produzir efeitos momentâneos; mas estruturar uma fala, ou um algoritmo, com potência formativa e transformadora requer técnica, estudo e treino.

A popularização do termo pensamento computacional, desde a formulação proposta por Wing (2006), tem gerado interpretações imprecisas sobre os elementos constitutivos da computação, sobretudo no que diz respeito à noção de algoritmo. Muitas vezes, essa noção é associada apenas a uma sequência lógica de ações humanas, descolando-se de sua função na modelagem de problemas computacionais por meio de estruturas. Para enfrentar essa imprecisão, vale recuperar a definição apresentada na Nota Técnica n.º 21 do Centro de Inovação para a Educação Brasileira - CIEB (2024), que busca recentrar o conceito em sua dimensão técnica e operatória:

A matemática fornece uma linguagem formal e universal. E por meio dela, a Computação constrói modelos chamados de algoritmos. Embora possam ser descritos em diferentes níveis de abstração, com linguagens específicas que focam apenas nos elementos essenciais e relevantes, os algoritmos são sequências finitas de passos que levam a um determinado resultado. Pode haver complicações, como sequências condicionais e laços de repetição, mas todos os passos são definidos e podem ser compreendidos por uma máquina. (CIEB, 2024, p. 9)

Ao enfatizar que algoritmos são modelos formalizáveis, que exigem clareza, estrutura e relação com as capacidades da máquina, essa definição se opõe frontalmente à noção imprecisa de que basta pensar como um cientista para resolver problemas. Mais do que uma habilidade geral de organização lógica, a formulação algorítmica exige mediação, abstração e domínio de estratégias de implementação. Isso é especialmente relevante em tempos de IA generativa, quando a aparência de simplicidade (escreva um *prompt*, obtenha um programa) pode ocultar a complexidade algorítmica que opera nos bastidores.

O CRA, ao explicitar essas engrenagens invisíveis do pensamento algorítmico, permite que o professor e o aluno “falem bem com a máquina”, e compreendam como e por que dizem o que dizem e, sobretudo, como poderiam dizer melhor. Ele transforma o improvisado em intencionalidade, a intuição em projeto, e o código em gesto autoral.

Crer que o domínio do *prompt* substitui o aprendizado da programação é uma forma atualizada de tecnodeterminismo, que transfere à máquina responsabilidades formativas que pertencem à consciência humana. Não se trata de negar as

potencialidades da IA. Ao contrário: é precisamente por reconhecer seu poder que se torna urgente a formação de sujeitos capazes de dialogar criticamente com ela.

Como já discutido na seção anterior, o Ciclo de Remediação Ativa (CRA) convida os alunos a ocuparem o papel de planejadores textual-algorítmicos. Essa postura se mostra ainda mais relevante diante de sistemas que mascaram a complexidade sob aparente simplicidade. Quem não entende as engrenagens, não sabe pedir. Ou, em outras palavras, quem não compreende o funcionamento interno do diálogo com a máquina delega mal e, ao fazê-lo, submete-se às soluções medianas que a IA pode oferecer.

A IA prevê. O humano inventa. A IA generaliza. O humano rompe padrões. A IA oferece o provável. Mas é o humano quem, com intenção, consciência e linguagem, pode propor o improvável.

Não se trata, portanto, de rejeitar os *prompts*, mas de qualificar sua elaboração como exercício de pensamento computacional, expressividade autoral e crítica algorítmica. O CRA, nesse sentido, emerge como um antídoto formativo à ilusão da simplicidade: uma proposta que forma sujeitos capazes de pensar antes de delegar, estruturar antes de improvisar, e projetar antes de solicitar.

Este ensaio não se opõe às investigações contemporâneas sobre engenharia de *prompts* ou às abordagens técnicas voltadas à otimização de interações com IA. Ao contrário, apresenta uma proposta voltada a explicitar os fundamentos formativos que tornam tais interações cognitivamente significativas no contexto educacional. Tais fundamentos formativos articulam três dimensões indissociáveis: consciência algorítmica, autoria linguística e metacognição crítica.

4. A Formação Docente e o CRA: vozes da prática

É no chão da formação, nos gestos dos sujeitos que experienciam o ciclo, que o CRA revela sua potência. Após os três meses de encontros e atividades formativas, um questionário foi aplicado a 16 professores articuladores do ensino fundamental vinculados a uma rede municipal de ensino, com aprovação ética registrada sob o CAAE 77530824.0.0000.5347.

As respostas abertas ao questionário foram analisadas por meio de uma abordagem qualitativa, com base na análise de conteúdo (Bardin, 2020), buscando identificar padrões de sentido recorrentes quanto à percepção da proposta, apropriação do ciclo e desafios enfrentados pelos docentes. A partir dessa análise, emergiram quatro eixos principais: encantamento, dificuldades, aprendizagens e sugestões. Essa sistematização empírica, ao dar voz aos professores em sua vivência do ciclo, sustenta a tese central deste ensaio: a de que o CRA é um gesto epistêmico, e não uma mera atividade de Scratch.

4.1. Integração que encanta: linguagem, tecnologia e autoria

Entre as primeiras palavras que emergem nas respostas dos docentes, nota-se o encantamento com a proposta. Esse encantamento, contudo, não é superficial, ele nasce do encontro entre dois mundos que, tradicionalmente, foram apartados na formação escolar: o da linguagem e o da computação: “Gostei da proposta de unir a fábula à

programação”; “A construção da narrativa no Scratch foi surpreendente”; “Nunca imaginei que eu conseguiria programar uma história escrita por mim.”

Essas vozes sinalizam um deslocamento importante. O CRA permitiu que os professores deixassem de ver a programação como algo alheio, técnico ou inacessível, e passassem a reconhecê-la como linguagem de expressão autoral. A fábula, nesse processo, atuou não apenas como gênero disparador, mas como estrutura mediadora: foi por meio dela que os sujeitos puderam transformar enredo em algoritmo, imagem em ação, imaginação em código.

Essa integração revelou-se formativa porque rompe com uma dicotomia profundamente enraizada na escola: a de que as humanidades e as tecnologias se opõem. No CRA, elas dialogam, se complementam e se reconfiguram.

4.2. As dificuldades que revelam o necessário: tempo e mediação

Apesar do entusiasmo, as respostas também revelam tensões, e essas tensões são valiosas. Os professores apontam dois desafios centrais: a lógica da programação e o tempo necessário para a apropriação do ciclo: “Tive dificuldade com os blocos”; “Faltou tempo para explorar com calma”; “Às vezes me perdia na lógica do que eu queria que acontecesse.”

As falas não indicam fracasso, mas sim uma percepção lúcida: é preciso tempo para pensar algoritmicamente. Esse tempo não é apenas cronológico, é também epistemológico. Aprender a planejar com clareza exige fricção, reescrita, ensaio e erro. Não há CRA sem mediação, sem pausas para refletir, sem devolutivas que provoquem deslocamentos. As dificuldades com os blocos do Scratch revelam menos um problema da ferramenta e mais um sinal de que a estrutura do pensamento algorítmico precisa ser construída, e isso não acontece em ritmo acelerado, mas sim por meio de processos formativos densos, reiterativos e mediados.

4.3. O que se aprende quando se estrutura o pensamento

Diante das dificuldades, emergem também conquistas profundas. Os professores aprenderam, sim, a usar um software necessário para finalização do ciclo, mas sobretudo aprenderam a pensar diferente. Há uma consciência crescente de que escrever um algoritmo é organizar ideias com clareza, intencionalidade e propósito: “Aprendi a organizar melhor as ideias”; “Percebi que escrever um algoritmo é como planejar um texto”; “Agora entendo que não é só colocar blocos, mas pensar antes o que se quer.”

Essas falas reiteram um dos princípios centrais do CRA: antes de codificar, é preciso planejar. Antes de escrever para a máquina, é preciso escrever para si mesmo. O algoritmo, como o texto, exige estrutura. E é nesse deslocamento que se realiza a formação, não no domínio técnico de uma linguagem, mas na consciência da lógica que a sustenta. Embora o Scratch tenha sido o ambiente utilizado na formação relatada, a aprendizagem no contexto do CRA não se restringe a esse ambiente de implementação. Ao experienciar o planejamento algorítmico na construção de narrativas interativas, os professores desenvolvem estruturas cognitivas que também fundamentam interações críticas com sistemas de IA generativa, como na formulação de *prompts* que explicitam condições, critérios e sequências, ou na avaliação reflexiva de respostas automatizadas.

Formular um *prompt* eficaz, interpretar uma resposta automática ou revisar um código sugerido por IA exige a mesma clareza estrutural mobilizada na organização de

condições, sequências e outras estratégias associadas ao pensamento computacional. Nesse sentido, a experiência com o CRA não instrumentaliza o professor para uma ferramenta específica, mas para uma postura diante da tecnologia: pensar antes de delegar, estruturar antes de solicitar e revisar antes de aceitar.

5. O CRA como estratégia contra o pensamento submisso

Se há algo que a ascensão das inteligências artificiais generativas deveria nos ensinar é que o lugar da programação não desapareceu, apenas se deslocou. A promessa de que bastaria escrever em linguagem natural para obter resultados mágicos não elimina a necessidade de pensamento algorítmico estruturado. Ao contrário: torna-o ainda mais necessário.

Aqueles que não compreendem as engrenagens do processo computacional continuam escrevendo, mas sem saber o que estão pedindo. Continuam usando a máquina, mas sem saber o que ela faz. Continuam se comunicando, mas sem saber como pensar. Essa delegação cega, travestida de praticidade, produz não sujeitos autônomos, mas usuários encantados, domesticados pela aparência de eficiência.

O Ciclo de Remediação Ativa (CRA) é, nesse cenário, uma estratégia de resistência epistemológica. Ele não forma promptistas tagarelas, mas planejadores críticos, capazes de refletir sobre o que pedem, por que pedem, e como o que pedem se traduz em lógica de execução. O CRA desloca o foco do fazer técnico para o pensar estratégico; da repetição do que já existe para a invenção do que ainda não foi modelado.

Como já defendia Papert (1980), o valor do pensamento computacional não está em decorar comandos, mas em formar modos de pensar sobre o próprio pensar. Programar, nesse sentido, não é fetiche técnico: é linguagem mental. É maneira de organizar ações, testar hipóteses, prever consequências, antecipar erros. É, sobretudo, um treino para a consciência.

Por isso, o CRA não é uma metodologia neutra, mas um gesto político: ele aposta na formação de sujeitos que não se deixam capturar pela lógica submissa da delegação total, mas que mantêm o controle reflexivo sobre suas próprias ideias, sobre os sistemas que utilizam, e sobre as mediações que os cercam.

6. Considerações finais

O Ciclo de Remediação Ativa (CRA) propõe-se como um gesto epistêmico: um modo de pensar e praticar a educação que resiste à automatização irrefletida da experiência escolar e convoca professores e estudantes a reocuparem, com autoria e consciência crítica, os territórios simbólicos da linguagem e da tecnologia. Em tempos em que o pensamento computacional é frequentemente reduzido a um rótulo genérico, esvaziado de seu potencial heurístico e algorítmico, o CRA opera como contramovimento, como tática de reinserção do planejamento, da mediação e da autoria nos processos educativos mediados por máquinas.

Ao propor a articulação entre leitura, reescrita, planejamento, prototipagem, programação e metacognição, o CRA desnaturaliza a delegação precoce da criação à inteligência artificial e reinscreve no coração da prática pedagógica a presença inegociável do sujeito pensante. Afinal, como sugerido ao longo deste ensaio, escrever

um bom *prompt*, construir um algoritmo ou compor um texto bem estruturado são faces distintas de uma mesma exigência: planejar antes de executar, elaborar antes de automatizar. Nesse sentido, o CRA desenvolve competências técnico-linguísticas e forma sujeitos capazes de escolher, julgar, adaptar e (re)significar, sujeitos que pensam antes de delegar.

As experiências formativas analisadas neste trabalho mostram que a integração entre linguagem e tecnologia pode ser encantadora, sim, mas também desafiadora. E é justamente nesse embate, entre encantamento e obstáculo, entre desejo e dificuldade, que emerge a necessidade de práticas docentes intencionais, coletivas e reflexivas. Quando docentes afirmam que “programar é como planejar um texto” ou que “escrever algoritmos ajuda a organizar ideias”, estão testemunhando o surgimento de um *ethos* pedagógico que recusa a separação entre saberes e reivindica a complexidade como valor formativo.

A escuta docente apresentada neste ensaio revela que o CRA, ao unir linguagem e lógica, desafia dicotomias arraigadas e abre espaço para novas sínteses: entre humanidades e computação, entre imaginação e rigor, entre autoria e técnica. Professoras e professores, ao reconhecerem o valor formativo do ciclo, apontam caminhos de apropriação que ultrapassam o “gostei” ou o “foi difícil”; eles delineiam uma gramática do encantamento pedagógico que nasce do esforço coletivo de pensar com as máquinas, e não por ou como elas.

Como desdobramento possível, vislumbra-se a criação de acervos escolares de narrativas digitais autorais, que combinem ferramentas de programação visual (como o Scratch) com recursos de inteligência artificial, respeitando os repertórios locais dos estudantes e potencializando práticas integradas de letramento, criatividade e pensamento computacional. Tais acervos não seriam apenas produtos, mas processos pedagógicos vivos, voltados à formação de sujeitos que saibam ler e escrever algoritmos com intencionalidade, autoria e consciência crítica.

Discutir formação algorítmica crítica na era da IA implica também reconhecer que tais tecnologias operam em redes sociotécnicas complexas, atravessadas por dimensões econômicas, ambientais e políticas. Seu funcionamento depende de infraestruturas materiais e cadeias de produção que não são neutras. Uma educação que se pretende crítica não pode ignorar esses aspectos. Contudo, situar essas condições não significa demonizar a tecnologia, mas compreendê-la como artefato histórico cuja apropriação pedagógica deve ser acompanhada de consciência e responsabilidade.

Por fim, trabalhar com CRA é trabalhar contra o pensamento submisso. É afirmar que o sujeito programador não é aquele que reproduz padrões, mas aquele que inventa caminhos. E é exatamente esse tipo de formação que o presente ensaio deseja fomentar: uma formação que una código e contexto, algoritmo e afeto, linguagem e justiça.

Uso de Inteligência Artificial

Ferramentas de inteligência artificial generativa foram utilizadas na elaboração deste artigo para gerar a Figura 1.

Referências

- Bardin, L. (2020). *Análise de Conteúdo*. Edições 70.
- Bolter, J. D., Grusin, R. (2000). *Remediation: Understanding new media*. MIT Press.
- Brasil. Ministério da Educação, Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. (2024). Saeb: Sistema de Avaliação da Educação Básica. <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/saeb>
- Brasil. Ministério da Educação. (2018). *Base Nacional Comum Curricular*. MEC.
- Brasil. Ministério da Educação, Conselho Nacional de Educação. (2022). Resolução CNE/CEB n. 1, de 4 de outubro de 2022: Normas sobre Computação na Educação Básica – Complemento à BNCC (2018).
- Centro de Inovação para a Educação Brasileira. (2024). Notas técnicas n. 21: Inteligência artificial na educação básica – novas aplicações e tendências para o futuro. https://cieb.net.br/wp-content/uploads/2024/06/Inteligencia-Artificial-na-Educacao-Basica_2024.pdf
- Computer Science Teachers Association. (2017). CSTA K–12 computer science standards (Revised). <https://www.csteachers.org/page/standards>
- Denning, P. J., Tedre, M. (2019). *Computational thinking*. MIT Press.
- Denning, P. J., Tedre, M. (2021). Computational thinking: A disciplinary perspective. In *Informatics in Education*, 20(1), 361–390. <https://doi.org/10.15388/infedu.2021.21>
- Dindler, C., Iversen, O. S., Caspersen, M. E., Smith, R. C. “Computational Empowerment”. In: Kong, Siu-Cheung; Abelson, Harold. *Computational Thinking Education in K–12: Artificial Intelligence Literacy and Physical Computing*. Cambridge: MIT Press, 2022. cap. 6. <https://doi.org/10.7551/mitpress/13375.003.0009>
- Faraco, C. A. (2015). *Pedagogia da variação linguística: Língua, diversidade e ensino*. Parábola Editorial.
- Foohs, M. M., Giraffa, L. (2022). Remediação do meio impresso para narrativas digitais: Uma proposta de metodologia ativa usando o Scratch. In *Educação em Revista*, 38, e35770. <https://doi.org/10.1590/0102-469835770>
- Foohs, M. M., Rebouças, R. O. M., Krüger, J. P. (2025). Pensamento computacional e sua identidade conceitual no pós-BNCC in *Revista Edutec - Educação, Tecnologias Digitais e Formação Docente*, 5(1). <https://doi.org/10.55028/edutec.v5i1.23231>
- Koch, I. V. (2020). *Introdução à linguística textual*. Contexto.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.
- Resnick, M. (2020). *Jardim de infância para a vida toda: por uma aprendizagem criativa, mão na massa e relevante para todos*. Penso.
- Rajo, R. (2012). *Letramentos múltiplos, escola e inclusão social*. Parábola Editorial.
- Vygotsky, L. S. (1998). *Pensamento e linguagem*. Martins Fontes.

Vygotsky, L. S. (2001). A formação social da mente: O desenvolvimento dos processos psicológicos superiores. Martins Fontes.

Wing, J. M. (2006). Computational thinking. In *Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>